

燃料电池发动机优化控制建模与元胞蚂蚁算法

王周緬^{1,2}, 徐晓明³, 马良¹

WANG Zhou-mian^{1,2}, XU Xiao-ming³, MA Liang¹

1. 上海理工大学 管理学院, 上海 200093

2. 武警上海政治学院, 上海 200435

3. 同济大学 汽车学院, 上海 200092

1. Management College, University of Shanghai for Science and Technology, Shanghai 200093, China

2. Armed Police Force Political Academy of Shanghai, Shanghai 200435, China

3. Automobile College, Tongji University, Shanghai 200092, China

WANG Zhou-mian, XU Xiao-ming, MA Liang. Application of cellular ant algorithm optimization controlling of fuel cell (PEMFC) engine. Computer Engineering and Applications, 2007, 43(19): 240-244.

Abstract: In this paper, NN is used to model the controlling system of fuel cell engine. And to shorten the temperature rising time and advance the output power of Fuel Cell Engine, a new evolutionary algorithm is applied to the optimization controlling of Fuel Cell Engine. Ant Colony Optimization (ACO) is a new optimization algorithm, and it has many good features. So it is widely applied to complication combinatorial optimization problems. But there is much deficiency. Specially, its searching speed is slow. And it will easily get into local optimization. This paper gives a new optimization algorithm which combines the principle of cellular automata with ant algorithm. The algorithm raises its searching speed with evolving the living state of every node. Simulation in Fuel Cell (PEMFC) Engine cold start controlling shows that the result is better than ant algorithm.

Key words: cellular ant algorithm; fuel cell (PEMFC) engine; NN modeling; optimization control

摘要: 以某燃料电池发动机为原型, 将神经网络辨识方法应用到其非线性系统的建模。在此基础上, 将元胞蚂蚁算法应用于燃料电池发动机优化控制问题, 以低温启动时暖机时间(升温至正常工作温度范围)最短和提高燃料电池发动机的输出功率为优化控制目标。蚂蚁算法是一种新型的进化算法, 它具有许多优良的性质, 因此被广泛用于求解各类组合优化问题。但基本的蚂蚁算法有许多不足, 比如它搜索速度慢。将元胞自动机与蚂蚁算法结合, 通过对结点的生死状态演变, 加快了算法的寻优速度, 并最大可能地避免陷入局部最优。最后通过对模型的测试, 采用元胞蚂蚁算法优化后的控制方法基本达到了上述目标, 证明了建模的合理性的同时, 显示出了该文优化控制方法的优越性。

关键词: 元胞蚂蚁算法; 燃料电池发动机; 神经网络建模; 优化控制

文章编号: 1002-8331(2007)19-0240-05 **文献标识码:** A **中图分类号:** TP391; TP18

1 引言

在当前的研究条件下, 要建立一个以燃料电池发动机硬件为基础的开发平台尚有很大的难度, 还需要做不少的准备工作。因此, 本文试图绕开 FCE 系统的内部复杂性, 将神经网络辨识方法应用到 FCE 非线性系统的建模, 尝试建立一个以软件仿真为基础的开发平台。

燃料电池发动机高效可靠运行的决定因素, 除燃料电池本身的制造工艺以外, 控制系统是最重要的因素。控制系统将根据负载对功率的要求, 随时调节反应气体的流量、压力达到最适当的数值, 以保证电化学反应有效地进行和反应产物水适时

地排出。通常情况下, 电池运行压力的提高有利于燃料电池性能的提高。特别是燃料电池发动机在低温启动时, 其电池温度不可能马上升高到其正常工作温度范围, 因此更有必要适当地提高燃料的供给压力。但是燃料电池消耗的空气需要燃料电池产出的电能带动风机来提供, 过高的压力必然要消耗更多的功率, 也就降低了燃料电池发动机的输出功率。

本文正是在这个背景下, 提出应用元胞蚂蚁算法解决燃料电池发动机的优化控制问题。在寻求提高压力以求尽快达到燃料电池发动机的正常工作温度范围的基础上还要考虑提高燃料电池发动机的输出功率。

基金项目: 国家高技术研究发展计划(863)(the National High-Tech Research and Development Plan of China under Grant No.2001AA501011); 国家自然科学基金(the National Natural Science Foundation of China under Grant No.70471065); 上海市重点学科建设资助项目(the Emphases Subject Constructing Foundation of Shanghai of China under Grant No.T0502)。

作者简介: 王周緬(1981-), 女, 上海理工大学管理学院博士研究生, 从事智能优化研究; 徐晓明(1980-), 男, 硕士, 研究方向: 汽车电子; 马良(1964-), 男, 教授, 博士, 博士生导师, 从事系统工程、智能优化研究。

2 燃料电池发动机原型描述

燃料电池发动机系统原型如图 1 所示。

(1) H₂ 供给系统

系统中采用高纯度、一定压力的 H₂, H₂ 罐出口处的压力传感器用来指示其内的含气量。高压 H₂ 经过减压/稳压阀后,再通过电动调节阀、压力传感器、流量计及加湿器进入电堆。

(2) 空气供给系统

高压风机将经过空气滤清器滤清的洁净空气的压力按照控制策略增压,然后经过压力传感器、空气加湿器进入电堆参与反应,多余的空气直接排入大气。

(3) 水/热管理系统

为保证电堆中电化学反应的正常高效进行,应严格控制它的温度,以确保其处于热平衡状态。通常是通过对冷却水温的控制来实现热平衡,如电堆进口水温 60℃,出口 65℃左右。

(4) 控制系统

通过各个传感器、ECU 和执行元件来完成对空气供给系统和水/热管理系统的控制,以满足负载变化对 FCE 功率的要求。

3 燃料电池发动机的神经网络建模

根据原型描述, H₂ 供给系统并不需要实时控制,因此可以认为 H₂ 的供给量和压力是恒定的。为了提高模型的准确性和便于分析,按照此原型本文先以电堆冷却水出水温度和冷却风扇转速为输入量,以冷却后的电堆冷却水进水温度响应为输出量,根据实验的输入输出数据,用神经网络辨识建立 FCE 系统的水/热管理模型。再以此电堆冷却水进水温度和风机频率为输入量,以 FCE 电堆的冷却水出水温度响应和功率响应为输出量。根据实验的输入输出数据,用神经网络辨识建立 FCE 系统的温度功率模型。

3.1 水/热管理模型

3.1.1 水/热管理系统描述

水/热管理模型可以描述为:

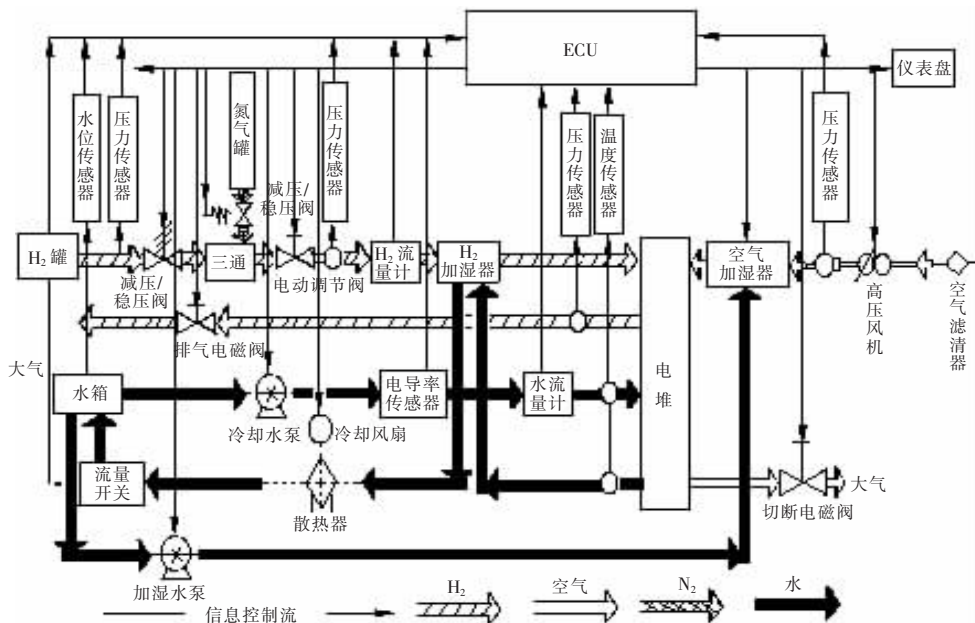


图 1 燃料电池发动机系统示意图

$$\frac{dT}{dt} = T_{in}(t) = \phi[T_{out}(t), V_{fan}(t)] \quad (1)$$

其中 $V_{fan}(t)$ 为控制输入量,即冷却风扇转速。 $T_{in}(t)$ 为输出量,即电堆冷却水进水温度。模型辨识的要求是在给定不同的电堆冷却水出水温度和冷却风扇转速下能够动态模拟出电堆冷却水进水温度的变化轨迹,即完成输入量到输出量的动态非线性映射。这样,辨识模型可用一个非线性差分方程描述为:

$$T_{in}(k+1) = \phi[T_{out}(k), V_{fan}(k)] \quad (2)$$

其中 k 是离散的时间变量。

3.1.2 水/热管理系统辨识的结构和算法

对水/热管理辨识的系统结构如图 2 所示。其中辨识采用的神经网络是含一个隐层的前馈 BP 神经网络,隐层神经元数为 36,传输函数是 S 型函数 $2/(1+\exp(-2^n)) - 1$,误差性能函数是平均方差。网络的训练样本数据集为 750 组不同情况下的温度动态响应数值,它存在于训练数据文件中,训练时可同时提供给网络。由于标准 BP 算法收敛速度较慢,因此训练时采用 Levenberg-Marquardt 的改进型 BP 算法。

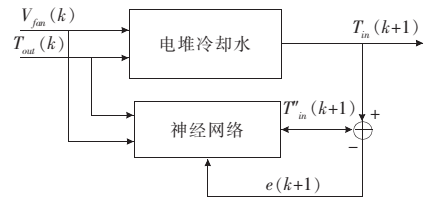


图 2 水/热管理辨识系统结构

3.1.3 水/热管理系统神经网络系统辨识结果

用上述训练好的神经网络对水/热管理系统进行辨识仿真,在各种情况下将神经网络计算的离散温度变化值与系统实际温度响应进行比较,辨识结果如图 3 所示。

由图 3 可以看出神经网络模型能够模拟系统的温度动态响应,最大误差不超过 0.8℃,误差是由训练时所确定的误差目标决定的。

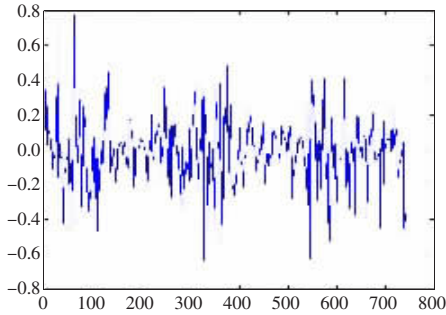


图3 离散温度变化值与系统实际温度响应误差

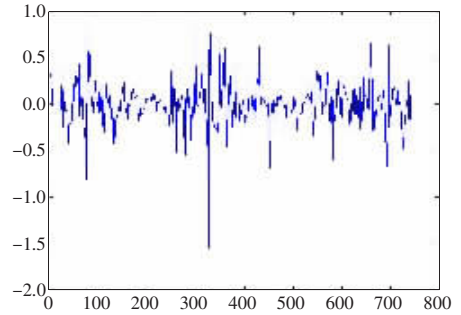


图5 离散温度变化值与系统实际温度响应误差

3.2 温度功率模型

3.2.1 FCE 系统的温度功率模型描述

温度模型可以描述为:

$$\frac{dT}{dt} = T_{out}(t) = \phi[T_{in}(t), F(t), T_{out}(t)] \quad (3)$$

功率模型可以描述为:

$$\frac{dP}{dt} = P_{out}(t) = \theta[T_{in}(t), F(t), T_{out}(t)] \quad (4)$$

其中 $F(t)$ 为控制输入量,即风机频率。 $T_{out}(t)$ 为输出量,即电堆冷却水出水温度。模型辨识的要求是在给定不同的电堆冷却水进水温度和风机频率下能够动态模拟出电堆冷却水出水温度和功率的变化轨迹,即完成输入量到输出量的动态非线性映射。这样,辨识模型可用两个非线性差分方程描述为:

$$T_{out}(k+1) = \phi[T_{in}(k), F(k), T_{out}(k)] \quad (5)$$

$$P_{out}(k+1) = \theta[T_{in}(k), F(k), T_{out}(k)] \quad (6)$$

其中 k 是离散的时间变量。

3.2.2 FCE 系统的温度功率系统辨识的结构和算法

对 FCE 系统的温度功率辨识的系统结构如图 4 所示。其中辨识采用的神经网络是含一个隐层的前馈 BP 神经网络,隐层神经元数为 40,传输函数是 S 型函数 $2/(1+\exp(-2^n)) - 1$,误差性能函数是平均方差。网络的训练样本数据集为 750 组不同情况下的温度、功率动态响应数值,它存在于训练数据文件中,训练时可同时提供给网络。训练时同样采用 Levenberg-Marquardt 的改进型 BP 算法。

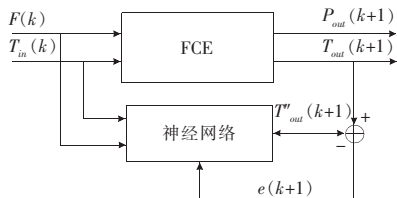


图4 FCE 系统的温度功率辨识系统结构

3.2.3 FCE 系统的温度功率神经网络系统辨识结果

用上述训练好的神经网络对 FCE 系统的温度功率进行辨识仿真,在各种情况下将神经网络计算的离散温度、功率变化值与系统实际温度、功率响应进行比较,辨识结果如图 5 和图 6 所示。

由图 5 和图 6 可以看出神经网络模型基本能模拟系统的温度、功率动态响应。最大温度误差不超过 1.6℃,最大功率误差不超过 2.5 kW。误差是由训练时所确定的误差目标决定的,可以通过多次试凑确定出最合适的神经网络拓扑结构以改善辨识精度。以上辨识过程表明采用神经网络辨识的方法实现 FCE 这类复杂非线性系统的建模是完全可行的。

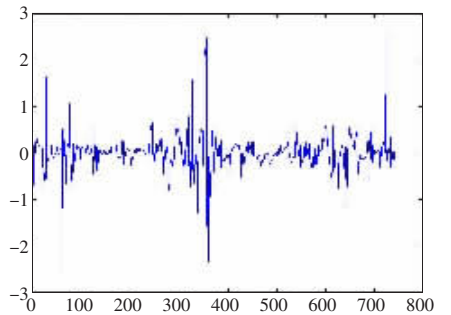


图6 离散功率变化值与实际功率响应误差

4 元胞蚂蚁算法的模型

蚂蚁算法是一种进化算法,它在 TSP 等典型的优化组合问题上表现了很好的寻优能力。这种算法是模仿蚂蚁间相互协作的方式:生物学家证明蚂蚁间的通信是通过信息素完成的,蚂蚁总是偏向于选择信息素浓的路径。元胞自动机是一种时间、空间、状态都离散,空间上相互作用及时间上的因果关系皆局部的网络动力学模型。其基本原理是利用大量元胞在简单的规则下的并行演化来模拟复杂而丰富的宏观现象。元胞蚂蚁算法是将本文提出的,将元胞自动机思想加入到蚂蚁算法所形成的一种新型的优化算法,它在保留了蚂蚁算法良好的寻优能力的同时,进一步避免了陷入局部最优。

以下是其基本模型:

(1)元胞单元的划分:将每个结点作为一个元胞。

(2)状态的编码:每个结点有 3 个状态:0,1,2。初始状态为 0;两结点状态均为 1 时表示在此次遍历中这两点间的路径需要进行信息素调整,两结点状态有一个不是 1 时都不需要调整这两点之间的信息素;状态为 2 时,不再演变此结点状态。

(3)邻居定义:两结点间有路径,且路径的权值小于某常数(此常数由权值矩阵确定)的各个点互定义为邻居。

(4)演化规则:

①初始化时所有结点状态置 1。

②当第 k 个蚂蚁转移到第 m 点上时,此结点的状态置为 1,与此结点的两个权值均小于某一常数的结点的生死状态置为 1。

③若某结点状态为 1,且在最近一次路径选择中被选择,则演变为 2,否则不变。

④若某结点状态已为 2,则不参与演变。

(5)信息素更新:

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$

其中 ρ 为参数, $(1-\rho)$ 表示在时刻 t 和 $t+n$ 之间信息素的蒸发率。

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k$$

$\Delta\tau_{ij}^k$ 是單位長度上在時刻 t 和 $t+n$ 之間第 k 個螞蟻在邊 $e(i,j)$ 留下的信息素數量;

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{如果在時刻 } t \text{ 和 } t+n \text{ 之間第 } k \text{ 個螞蟻使用邊 } e(i,j) \\ 0 & \text{其他} \end{cases}$$

其中, Q 是一個常數, L_k 是第 k 個螞蟻周遊的路程長度。

(6) 第 k 個螞蟻從城市 i 到城市 j 的轉移概率為:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & \text{if } j \in allowed_k \\ 0 & \text{其他} \end{cases}$$

其中, $allowed_k = \{N - tabu_k\}$, α 和 β 是控制信息素與可見度的相對重要性的參數。轉移概率是可見度和 t 時刻信息素強度的權衡。 η_{ij} 為邊弧 (i,j) 的能見度 (visibility), 即 $1/d_{ij}$ 。

5 元胞螞蟻算法應用於燃料電池發動機控制的算法主要步驟

- Step1 $A \leftarrow \{\text{所有工作溫度、功率狀態下的原始網絡圖集合}\}$; 按溫度對 A 進行排序;
- Step2 $C := \{\text{低於當前工作溫度狀態下的網絡圖集合}\}; (C \in A)$
 $A \leftarrow (A - C);$ //更新 A 中各網絡圖
- Step3 $B := \text{Algorithm1}(A);$ //針對當前工作溫度狀態定義預求解網絡集
 $A \leftarrow (A - B);$ //更新 A 中各網絡圖
- Step4 $B := \text{Algorithm2}(B)$ //對 B 中各網絡用算法 2 進行求解, 求出最短路徑
 $L \leftarrow L_{best};$ //保存此最短路徑
 If $A \neq \emptyset$ then go to Step3;
- Step5 $L_f \leftarrow (L);$ //求出最短增壓路徑

5.1 Algorithm1 算法描述

- $T \leftarrow (net_{temp});$ //更新當前工作溫度
- $T_1, T_2;$ //預定義工作溫度
- $\Delta T_1, \Delta T_2, \Delta T_3;$ //預定義工作溫度增量
- if $(T < T_1)$ then begin $\Delta T = \Delta T_1; B \leftarrow A(\Delta T, T_1, T_1)$ end;
- else if $(T_1 < T < T_2)$ then begin $\Delta T = \Delta T_2; B \leftarrow A(\Delta T, T_1, T_2)$ end;
- else begin $\Delta T = \Delta T_3; B \leftarrow A(\Delta T, T_2)$ end

5.2 Algorithm2 算法描述

- (1) Initialization phase
- $t \leftarrow 0$ (t 為時刻變量)
- $nc \leftarrow 0$ (nc 為迭代次數)
- 對各邊 $E(i,j)$: 置 $\tau_{ij} \leftarrow$ 常數 C (較小的正數)
 置 $\Delta\tau_{ij} \leftarrow 0$
 確定所有點的響應半徑
 建立各點的鄰居集合
- L_{best} 置空 (用於儲存當前最好的路徑隊列)
- $L_m \leftarrow$ 常數 a (a 為較大的正數, L_m 用來儲存當前最短的增壓路徑長度)
- T_0 : 初始點的溫度 P_0 : 初始點的功率

T_{set} : 給定點的溫度 P_{set} : 給定點的功率

P : 消耗的功率 T 所用的時間和

(2) this is the phase in which ants build their routes, the route of ant in stored in L_k

將 m 個螞蟻放在初始點上

for $k \leftarrow 1$ to m do

begin 初始化禁忌表; 置空路徑隊列 L

while 給定點不在禁忌表中

begin 按轉移概率 $p_{ij}^k(t)$ 選擇下一結點 V_j (j not in 禁忌表)

判斷 V_j 的生死值, 生則 go on, 死則重新選擇 V_j

V_j 以及與 V_j 同溫度的所有點加入禁忌表

禁忌表外的各點的所有邊進行有限次生死演變

路徑 E_{ij} 置於隊列 L_k end{end while};

計算螞蟻 k 的兩個目標函數 T, P

記錄當前的非劣解

end {end for }

(3) in this phase global updating occurs and pheromone is updated

for $k \leftarrow 1$ to m do begin 對各邊弧 (i,j) 計算: $\Delta\tau_{ij} \leftarrow \Delta\tau_{ij} + \Delta\tau_{ij}^k$; end;

對各邊弧 (i,j) , 計算: $\tau_{ij}(t+n) \leftarrow \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$ (其中 $\tau_{min} \leq \tau_{ij} \leq \tau_{max}$)

置 $t \leftarrow t+n$;

對各邊弧 (i,j) , 置 $\Delta\tau_{ij} \leftarrow 0$;

$nc \leftarrow nc+1$;

//in this phase, end condition will be given

if $nc <$ 預定的迭代次數且無退化行為 (即找到的都是相同解) then goto phase (2)

6 FCE 的元胞螞蟻算法優化控制系統及其仿真

FCE 的元胞螞蟻算法優化控制系統的结构如图 7 所示。

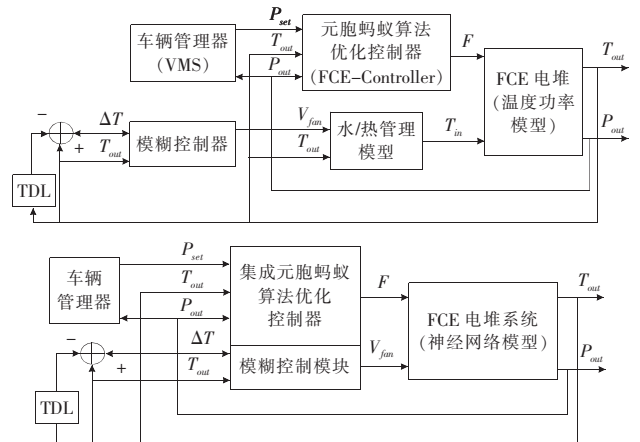


圖 7 FCE 的元胞螞蟻算法優化控制系統

應用該系統對 FCE 電堆溫度和功率進行仿真, FCE 電堆的溫度和功率動態特性由神經網絡辨識模型代替, 在線計算時, 採用了上節介紹的神經網絡辨識模型, 以獲得較快的特性輸出。其中冷卻風扇由現有的模糊控制器進行控制, 車輛管理器根據當前 FCE 輸出的功率 (P_{out}) 以及負載變化的情況給定車輛所需的功率 (P_{set})。

各結點坐標如表 1。

仿真測試結果如圖 8、圖 9 所示。

表1 控制点的各点坐标

	a	b	c	d	e	f	g	h	i	j	k	l	m
发动机内部温度/°C	24.5	25	25.5	26	26.5	27	27.5	28	28.5	29	29.5	30	30.5
发动机输出功率/kW	4.8	5.6	5.6	6.4	6.4	8	8	10	10	10	12	12	12
	n	o	p	q	r	s	t	u	v	w	x	y	z
发动机内部温度/°C	31	31.5	32	32.5	33	33.5	34	34.5	35	36	37	38	39
发动机输出功率/kW	15	15	15	18	18	18	21	21	21	24	24	24	27
	A	B	C	D	E	F	G	H	I	J	K	L	M
发动机内部温度/°C	40	41	42	43	44	45	46	47	48	49	50	51	52
发动机输出功率/kW	27	27	30	30	30	30	33	33	33	36	36	36	36
	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
发动机内部温度/°C	53	54	55	56	57	58	59	60	61	62	64	64	65
发动机输出功率/kW	38	38	38	38	40	40	40	40	42	42	42	44	44

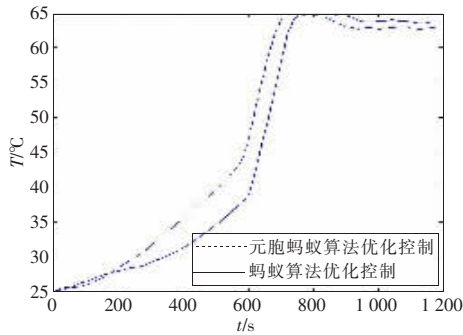


图8 低温启动元胞蚂蚁算法优化控制温度响应

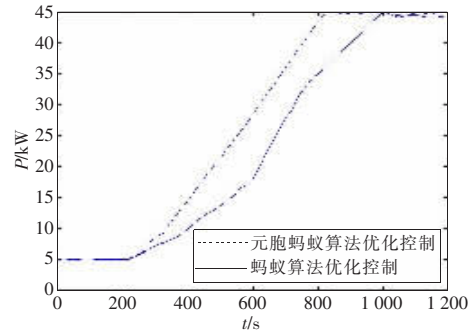


图9 低温启动元胞蚂蚁算法优化控制功率响应

7 结论

本文提出的元胞蚂蚁算法是基于元胞自动机原理和蚂蚁算法,利用元胞演化规则确定各结点的生死状态,使蚂蚁集中于某些区域,加快了搜索速度。通过其在燃料电池发动机优化控制上的应用,证明了该算法比蚂蚁算法具备比传统算法更快的收敛速度。(收稿日期:2007年3月)

(上接 239 页)

5 结论

本文采取约束条件下,多目标函数优化方法,结合惩罚函数,运用量子粒子群算法对易逝品的多目标最优定价进行了研究。对结果的分析表明,得到的最优定价符合商品经济规律。同时,仿真结果也为销售商在不同情况下确定最优定价提供了决策依据。

另外,运用量子粒子群算法,可以利用较少的参数设置,得到可靠的数据结果,显示了量子粒子群算法在全局寻优中的有效性和卓越性。(收稿日期:2007年3月)

参考文献:

- [1] Hwang C L, Yoon K. Multiple attributes decision making methods and applications[M]. Berlin Heidelberg: Springer, 1981.
- [2] Nahmias S. Perishable inventory theory: review[J]. Operations Research, 1982, 30: 680-780.
- [3] Weatherford L R, Bodily S E. A taxonomy and research overview of perishable-asset revenue management: yield management, overbooking,

参考文献:

- [1] 郭航, 马重芳. 汽车用质子交换膜燃料电池性能实验研究[J]. 中国公路学报, 2001, 14(4).
- [2] 胡里清, 李拯. 燃料电池运行压力对整车燃料效率的影响[J]. 电源技术, 2004, 128(5).
- [3] 马良, 蒋馥. 度限制最小树的蚂蚁算法[J]. 系统工程学报, 1999, 14(3).
- [4] 马良, 蒋馥. 多目标旅行售货员问题的蚂蚁算法求解[J]. 系统工程理论方法应用, 1999, 8(4).
- [5] and pricing[J]. Operations Research, 1992, 40(5): 831-844.
- [6] Winterfeldt D V, Edwards W. Decision analysis and behavioral research[M]. [S.l.]: Cambridge University Press, 1998.
- [7] Hepu D, Chung H Y, Robert J. Inter-company comparison using modified TOPSIS with objective weights[J]. Computers & Operations Research, 2000, 27: 963-973.
- [8] Young H Chun. Optimal pricing and ordering policies for perishable commodities[J]. European Journal of Operational Research, 2003, 144(1): 68-82.
- [9] Kennedy J, Eberhart R C. Particle swarm optimization[C]//Proceeding of the 1995 IEEE International Conference on Neural Networks, Perth, Australia, 1995: 1942-1948.
- [10] Sun J, Xu W B. A global search strategy of Quantum-behaved Particle Swarm Optimization[C]//Proceedings of IEEE Conference on Cybernetics and Intelligent Systems, 2004: 111-116.
- [11] Parsopoulos K E, Vrahatis M N. Particle swarm optimization method for constrained optimization problems[C]//Proceedings of the 2002 Euro-International Symposium on Computation Intelligence, 2002: 214-220.